# Personel sınıfları:

**Personel ana sayfası:**

public class PersonelAnasayfaController : Controller

{

Otopark\_OtomasyonuEntities1 db = new Otopark\_OtomasyonuEntities1();

// GET: PersonelAnasayfa public ActionResult Index()

{

return View(db.Parkdurumu.ToList());

}

}

# Personel login girişi

public class PersonelGirişController : Controller

{

Otopark\_OtomasyonuEntities1 db = new Otopark\_OtomasyonuEntities1();

//private object FormsAuthentication;

// GET: PersonelGiriş public ActionResult Login()

{

return View();

}

[HttpPost]

public ActionResult Login(Personel\_Giriş Personel)

{

var personel = db.Personel\_Giriş.FirstOrDefault(x => x.TC\_No == Personel.TC\_No && x.Sifre == Personel.Sifre);

if (personel != null)

{

FormsAuthentication.SetAuthCookie(personel.TC\_No , false); return RedirectToAction("Index", "PersonelAnasayfa");

}

else

{

ViewBag.Mesaj = "Geçersiz Kullanıcı Adı veya Şifre"; return View();

}

}

public ActionResult Logout()

{

FormsAuthentication.SignOut(); return View("Login");

}

}

# Personel rezervasyon bilgileri

public class PerRezervasyonController : Controller

{

private Otopark\_OtomasyonuEntities1 db = new Otopark\_OtomasyonuEntities1();

// GET: PerRezervasyon public ActionResult Index()

{

var rezervasyon = db.Rezervasyon.Include(r => r.Parkdurumu); return View(rezervasyon.ToList());

}

}

# Müşteri listesi

public class MusteriListesiController : Controller

{

private Otopark\_OtomasyonuEntities1 db = new Otopark\_OtomasyonuEntities1();

// GET: MusteriListesi public ActionResult Index()

{

return View(db.Musteri\_Listesi.ToList());

}

// GET: MusteriListesi/Details/5 public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Musteri\_Listesi musteri\_Listesi = db.Musteri\_Listesi.Find(id); if (musteri\_Listesi == null)

{

return HttpNotFound();

}

return View(musteri\_Listesi);

}

// GET: MusteriListesi/Create public ActionResult Create()

{

return View();

}

// POST: MusteriListesi/Create

// To protect from overposting attacks, please enable the specific properties you want to bind to, for

// more details see [http://go.microsoft.com/fwlink/?LinkId=317598.](http://go.microsoft.com/fwlink/?LinkId=317598) [HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "id,Tc\_No,Ad,Soyad,Cep\_Telefonu,Plaka,Kullanıcı\_Adı,E\_Posta,Sifre,Sifre\_Tekrar")] Musteri\_Listesi musteri\_Listesi)

{

if (!ModelState.IsValid)

{

return View("Create");

}

db.Musteri\_Listesi.Add(musteri\_Listesi); db.SaveChanges();

return RedirectToAction("Index");

//return View(musteri\_Listesi);

}

// GET: MusteriListesi/Edit/5 public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Musteri\_Listesi musteri\_Listesi = db.Musteri\_Listesi.Find(id); if (musteri\_Listesi == null)

{

return HttpNotFound();

}

return View(musteri\_Listesi);

}

// POST: MusteriListesi/Edit/5

// To protect from overposting attacks, please enable the specific properties you want to bind to, for

// more details see [http://go.microsoft.com/fwlink/?LinkId=317598.](http://go.microsoft.com/fwlink/?LinkId=317598) [HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include = "id,Tc\_No,Ad,Soyad,Cep\_Telefonu,Plaka,Kullanıcı\_Adı,E\_Posta,Sifre,Sifre\_Tekrar")] Musteri\_Listesi musteri\_Listesi)

{

if (ModelState.IsValid)

{

db.Entry(musteri\_Listesi).State = EntityState.Modified; db.SaveChanges();

return RedirectToAction("Index");

}

return View(musteri\_Listesi);

}

// GET: MusteriListesi/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Musteri\_Listesi musteri\_Listesi = db.Musteri\_Listesi.Find(id); if (musteri\_Listesi == null)

{

return HttpNotFound();

}

return View(musteri\_Listesi);

}

// POST: MusteriListesi/Delete/5 [HttpPost, ActionName("Delete")] [ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

Musteri\_Listesi musteri\_Listesi = db.Musteri\_Listesi.Find(id); db.Musteri\_Listesi.Remove(musteri\_Listesi); db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

# Müşteri araç giriş-çıkış işlemleri

public class Araç\_Giriş\_ÇıkışController : Controller

{

private Otopark\_OtomasyonuEntities1 db = new Otopark\_OtomasyonuEntities1();

// GET: Araç\_Giriş\_Çıkış public ActionResult Index()

{

var park\_Kayıt = db.Park\_Kayıt.Include(p => p.Parkdurumu); return View(park\_Kayıt.ToList());

}

// GET: Araç\_Giriş\_Çıkış/Details/5 public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Park\_Kayıt park\_Kayıt = db.Park\_Kayıt.Find(id); if (park\_Kayıt == null)

{

return HttpNotFound();

}

return View(park\_Kayıt);

}

// GET: Araç\_Giriş\_Çıkış/Create public ActionResult Create()

{

ViewBag.Park\_Yeri = new SelectList(db.Parkdurumu, "id", "parkno"); return View();

}

// POST: Araç\_Giriş\_Çıkış/Create

// To protect from overposting attacks, please enable the specific properties you want to bind to, for

// more details see [http://go.microsoft.com/fwlink/?LinkId=317598.](http://go.microsoft.com/fwlink/?LinkId=317598) [HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "id,Tc\_No,Ad,Soyad,Plaka,Park\_Yeri,Giriş\_Zamanı,Çıkış\_Zamanı,Ücret")] Park\_Kayıt park\_Kayıt)

{

if (ModelState.IsValid)

{

db.Park\_Kayıt.Add(park\_Kayıt); db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.Park\_Yeri = new SelectList(db.Parkdurumu, "id", "parkno", park\_Kayıt.Park\_Yeri);

return View(park\_Kayıt);

}

// GET: Araç\_Giriş\_Çıkış/Edit/5 public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Park\_Kayıt park\_Kayıt = db.Park\_Kayıt.Find(id);

if (park\_Kayıt == null)

{

return HttpNotFound();

}

ViewBag.Park\_Yeri = new SelectList(db.Parkdurumu, "id", "parkno", park\_Kayıt.Park\_Yeri);

return View(park\_Kayıt);

}

// POST: Araç\_Giriş\_Çıkış/Edit/5

// To protect from overposting attacks, please enable the specific properties you want to bind to, for

// more details see [http://go.microsoft.com/fwlink/?LinkId=317598.](http://go.microsoft.com/fwlink/?LinkId=317598) [HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include = "id,Tc\_No,Ad,Soyad,Plaka,Park\_Yeri,Giriş\_Zamanı,Çıkış\_Zamanı,Ücret")] Park\_Kayıt park\_Kayıt)

{

if (ModelState.IsValid)

{

db.Entry(park\_Kayıt).State = EntityState.Modified; db.SaveChanges();

return RedirectToAction("Index");

}

ViewBag.Park\_Yeri = new SelectList(db.Parkdurumu, "id", "parkno", park\_Kayıt.Park\_Yeri);

return View(park\_Kayıt);

}

// GET: Araç\_Giriş\_Çıkış/Delete/5 public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Park\_Kayıt park\_Kayıt = db.Park\_Kayıt.Find(id); if (park\_Kayıt == null)

{

return HttpNotFound();

}

return View(park\_Kayıt);

}

// POST: Araç\_Giriş\_Çıkış/Delete/5 [HttpPost, ActionName("Delete")] [ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

Park\_Kayıt park\_Kayıt = db.Park\_Kayıt.Find(id); db.Park\_Kayıt.Remove(park\_Kayıt); db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}